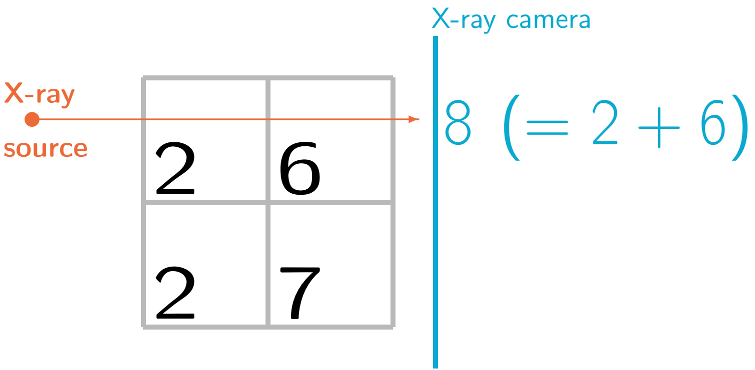
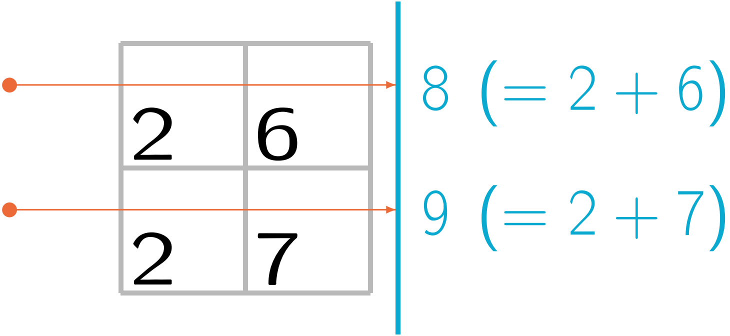
X-ray tomography minicourse: Monday exercises

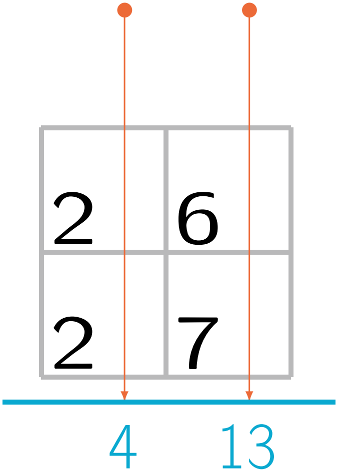
Samuli Siltanen

PhD Winter School
Advanced methods for mathematical image analysis
Bologna, January 23, 2023

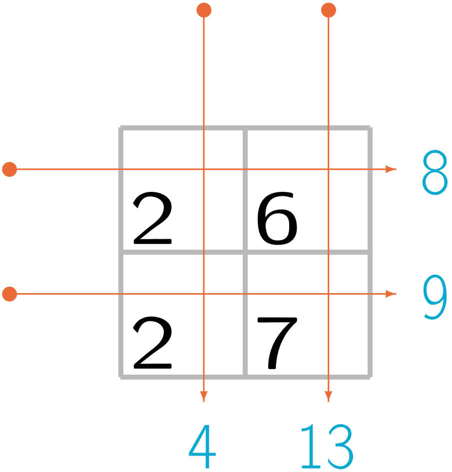
Outline


X-ray
source
2
6
2
7
X-ray camera
8 (= 2 + 6)

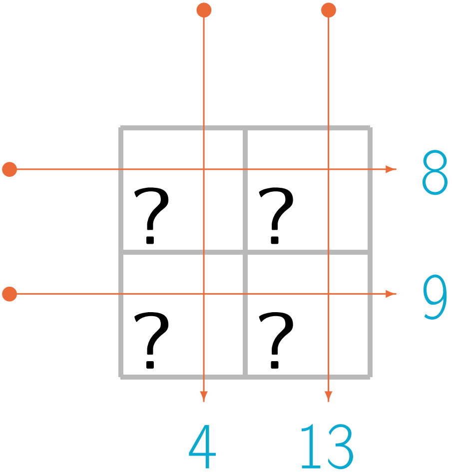

2
6
2
7
8 (= 2 + 6)
9 (= 2 + 7)



2
6
2
7
4
13



2
6
2
7
8
9
4
13


?
?
?
?
8
9
4
13

	4	13
8	2	6
9	2	7

	4	13
8	-1	9
9	5	4

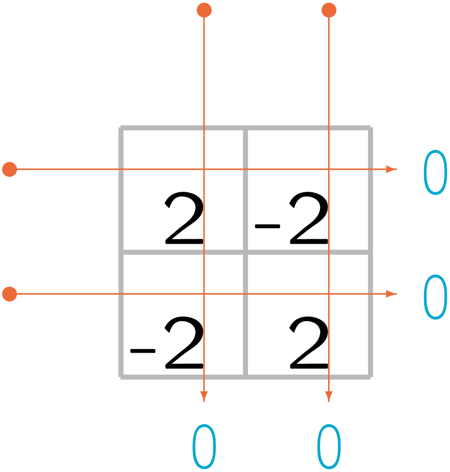
	4	13
8	2	6
9	2	7

	4	13
8	-1	9
9	5	4

$$\begin{array}{cc|c} 2 & 6 & 8 \\ 2 & 7 & 9 \\ \hline 4 & 13 & \end{array}$$

$$\begin{array}{cc|c} -1 & 9 & 8 \\ 5 & 4 & 9 \\ \hline 4 & 13 & \end{array}$$

$$\begin{array}{cc|c} 4 & 4 & 8 \\ 0 & 9 & 9 \end{array}$$



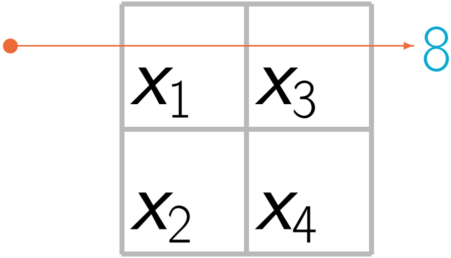
2
-2
-2
2
0
0
0
0

$$\begin{array}{cc} \begin{bmatrix} 2 & 6 \\ 2 & 7 \end{bmatrix} & \begin{matrix} 8 \\ 9 \end{matrix} \\ \begin{matrix} 4 & \quad 13 \end{matrix} & \end{array} + \begin{array}{cc} \begin{bmatrix} 2 & -2 \\ -2 & 2 \end{bmatrix} & \begin{matrix} 0 \\ 0 \end{matrix} \\ \begin{matrix} 0 & \quad 0 \end{matrix} & \end{array}$$

$$= \begin{array}{cc} \begin{bmatrix} 4 & 4 \\ 0 & 9 \end{bmatrix} & \begin{matrix} 8 \\ 9 \end{matrix} \\ \begin{matrix} 4 & \quad 13 \end{matrix} & \end{array}$$

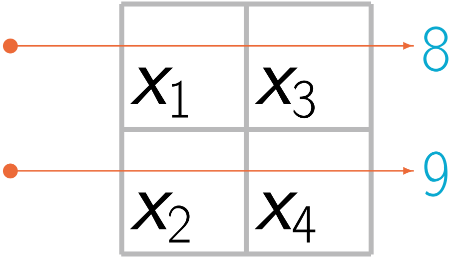
Outline

Each data point gives rise to one row in the measurement matrix



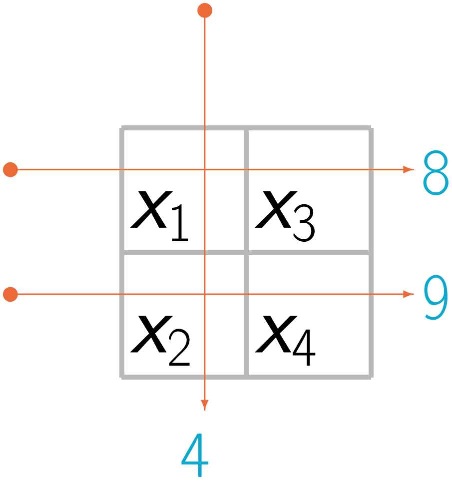
$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ & & & \\ & & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 8 \\ \\ \end{bmatrix}$$

Each data point gives rise to one row in the measurement matrix



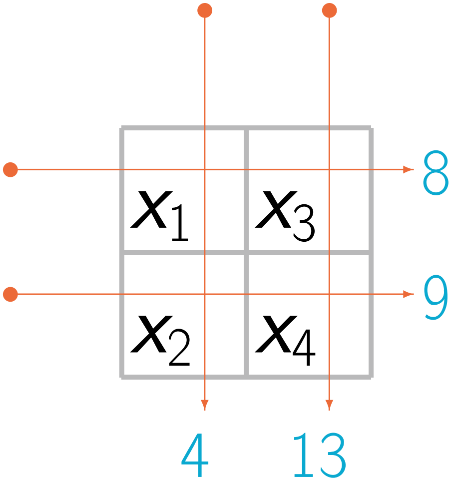
$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ & & & \\ & & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 8 \\ 9 \\ \\ \end{bmatrix}$$

Each data point gives rise to one row in the measurement matrix



$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ & & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 8 \\ 9 \\ 4 \\ \end{bmatrix}$$

Each data point gives rise to one row in the measurement matrix



$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 8 \\ 9 \\ 4 \\ 13 \end{bmatrix}$$

$$Ax = m$$

Outline

First 2×2 exercise

Determine the kernel of the measurement matrix

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

How is the kernel related to "ghosts", or objects that are nontrivial but give zero measurement?

Outline

- **Tomography with 2×2 pixels: non-uniqueness**
  - Matrix model for the measurement
  - First 2×2 exercise
  - Total variation regularization
  - Second 2×2 exercise
  - Numerical implementation
  - Third 2×2 exercise
- **Tomography with 1×2 pixels: ill-posedness**
- **Exercises, collected**

Let's study the two penalties used in regularization. We focus on three examples

Original patient

2	6
2	7

Flat candidate

3	3
3	3

Wrong data,
good "tissue type"

Spooky candidate

4	4
0	9

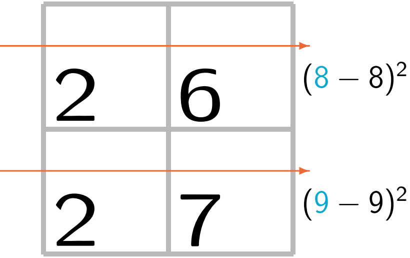
Correct data,
bad "tissue type"

Calculate data penalty for the original phantom

2	6
2	7

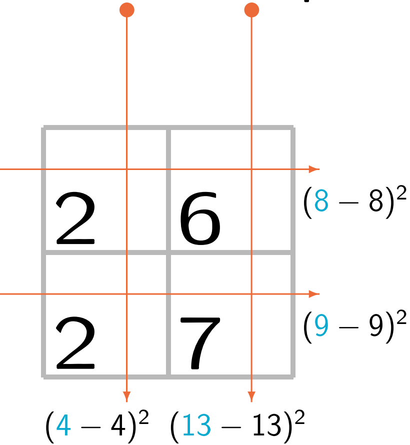
$(8 - 8)^2$

Calculate data penalty for the original phantom



Data penalty: $(8-8)^2 + (9-9)^2$

Calculate data penalty for the original phantom



Data penalty: $(8-8)^2 + (9-9)^2 + (4-4)^2 + (13-13)^2 = 0.$

Calculate prior penalty for the original phantom

2	6
2	7

Prior penalty: $|2-6|$

Calculate prior penalty for the original phantom

2	6
2	7

Prior penalty: $|2-6| + |2-7|$

Calculate prior penalty for the original phantom

2	6
2	7

Prior penalty: $|2-6|+|2-7|+|2-2|$

Calculate prior penalty for the original phantom

2	6
2	7

Prior penalty: $|2-6|+|2-7|+|2-2|+|6-7|=4+5+0+1=10.$

Total penalty is the sum of data&prior penalties

2	6
2	7

	data penalty	0
+	prior penalty	10
=	total penalty	€10

Data penalty for flat candidate

3	3	$(6-8)^2$
3	3	$(6-9)^2$
$(6-4)^2$	$(6-13)^2$	

Data penalty: $2^2 + 3^2 + 2^2 + 7^2 = 4 + 9 + 4 + 49 = 66.$

Prior penalty for flat candidate

3	3
3	3

Prior penalty: $|3-3|+|3-3|+|3-3|+|3-3|=0.$

Total penalty for flat candidate

3	3
3	3

	data penalty	66
+	prior penalty	0
=	total penalty	€66

Data penalty for spooky candidate

$(8-8)^2$	4	4
$(9-9)^2$	0	9
	$(4-4)^2$	$(13-13)^2$

Data penalty: $(8-8)^2 + (9-9)^2 + (4-4)^2 + (13-13)^2 = 0$.

Prior penalty for spooky candidate

4	4
0	9

Prior penalty: $|4-4|+|0-9|+|4-0|+|4-9| = 0+9+4+5 = 18.$

Comparison of the three candidates

Original patient

2	6
2	7

data penalty 0
+ prior penalty 10
= total penalty €10

Flat candidate

3	3
3	3

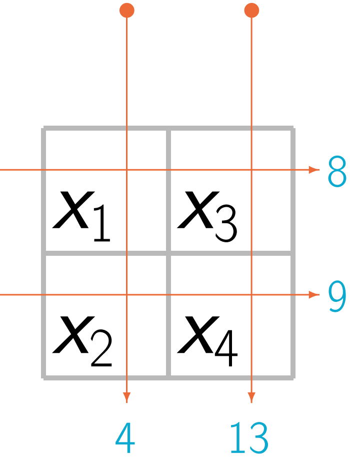
data penalty 66
+ prior penalty 0
= total penalty €66

Spooky candidate

4	4
0	9

data penalty 0
+ prior penalty 18
= total penalty €18

In practice we do not have three candidates. We need a general reconstruction algorithm



Find numbers $x_1 \geq 0$, $x_2 \geq 0$, $x_3 \geq 0$ and $x_4 \geq 0$ such that the sum of these two penalties is as small as possible:

Data penalty: $(x_1+x_3-8)^2+(x_2+x_4-9)^2$
$+(x_1+x_2-4)^2+(x_3+x_4-13)^2$

Prior penalty: $|x_1-x_3|+|x_2-x_4|$
$+|x_1-x_2|+|x_3-x_4|$

This method is called (anisotropic) **total variation regularization.**

The minimizer of the TV penalty functional has two "internal organs", as does the original

Original patient

2	6
2	7

TV minimizer

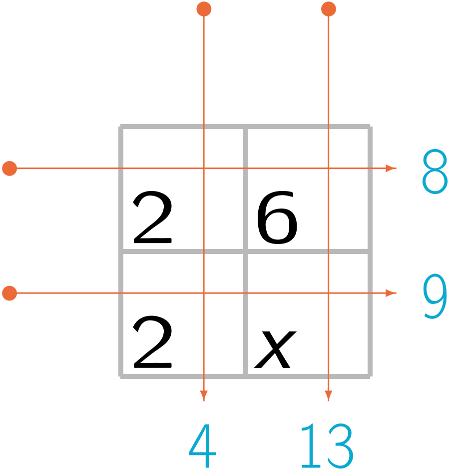
$2\frac{1}{4}$	$6\frac{1}{4}$
$2\frac{1}{4}$	$6\frac{1}{4}$

	Original patient	TV minimizer
data penalty	0	1
+ prior penalty	10	8
= total penalty	€10	€9

Outline

Second 2×2 exercise, slide 1/2

Assume that we know three pixel values and look for the fourth one, called x.



Second 2×2 exercise, slide 2/2

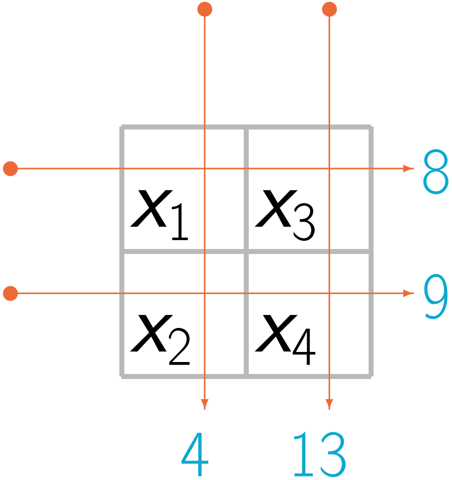
Take $\alpha = 1$. Write down the total variation penalty functional in the form

$$\widetilde{x} = \arg\min_{x \in \mathbb{R}} \{f(x)\}.$$

- Give the formula for f.
- Plot $f(x)$.
- At what points does f fail to be differentiable?
- Find the minimizing argument $\widetilde{x} \in \mathbb{R}$ approximately. You can either use brute-force forking or apply an optimization method.

Outline

Recall the matrix measurement model



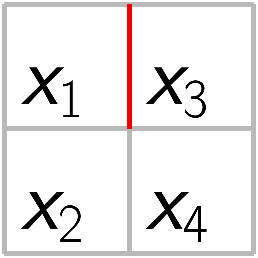
$$\begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} 8 \\ 9 \\ 4 \\ 13 \end{bmatrix}$$

$$Ax = m$$

We can now formulate (anisotropic) total variation regularization mathematically

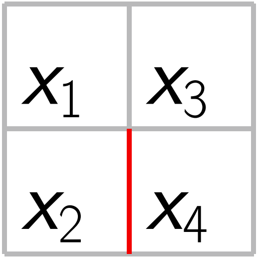
$$x_{\text{TV}} = \arg\min_{x \in \mathbb{R}^4} \{ \|Ax - m\|_2^2 + \|L_H x\|_1 + \|L_V x\|_1 \}$$

Writing the prior penalty in matrix form: construction of the horizontal difference matrix L_H



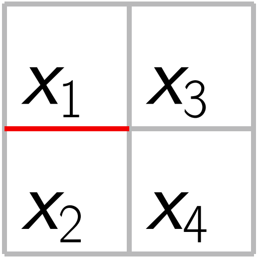
$$\begin{bmatrix} 1 & 0 & -1 & 0 \\ & & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 - x_3 \\ \\ \end{bmatrix}$$

Writing the prior penalty in matrix form: construction of the horizontal difference matrix L_H



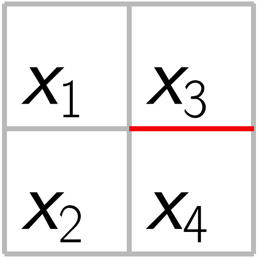
$$\underbrace{\begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix}}_{L_H} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 - x_3 \\ x_2 - x_4 \end{bmatrix}$$

Writing the prior penalty in matrix form: construction of the vertical difference matrix L_V



$$\begin{bmatrix} 1 & -1 & 0 & 0 \\ & & & \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 - x_2 \\ \end{bmatrix}$$

Writing the prior penalty in matrix form: construction of the vertical difference matrix L_V



$$\underbrace{\begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 \end{bmatrix}}_{L_V} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} = \begin{bmatrix} x_1 - x_2 \\ x_3 - x_4 \end{bmatrix}$$

Matrix formulation of the anisotropic total variation prior penalty

Our minimization problem:

$$x_{\text{TV}} = \arg\min_{x \in \mathbb{R}^4} \{ \|Ax - m\|_2^2 + \|L_H x\|_1 + \|L_V x\|_1 \}$$

Recall that for a vector $y \in \mathbb{R}^n$ we have $\|y\|_1 = |y_1| + |y_2| + \cdots + |y_n|$.

Therefore, the prior penalty can be written as (why? check!)

$$\begin{aligned} \|L_H x\|_1 + \|L_V x\|_1 \;\; = \;\; & |x_1 - x_3| + |x_2 - x_4| \\ & + |x_1 - x_2| + |x_3 - x_4|. \end{aligned}$$

Reformulation as a quadratic problem

We want to minimize the non-quadratic functional

$$\|Ax - m\|_2^2 + \|L_H x\|_1 + \|L_V x\|_1$$

over non-negative image vectors $x \in \mathbb{R}^4$. This task can be converted into minimizing the quadratic functional

$$\frac{1}{2} z^T Q z + c^T z$$

over non-negative $z \in \mathbb{R}^{12}$ with equality constraints $Ez = b$.

Rewriting the TV regularization using the trick of non-negative vectors

Write the horizontal and vertical differences in the form

$$L_{\mathsf{H}}x = u_{\mathsf{H}}^{+} - u_{\mathsf{H}}^{-} \quad \text{and} \quad L_{\mathsf{V}}x = u_{\mathsf{V}}^{+} - u_{\mathsf{V}}^{-},$$

using **non-negative** vectors $u_{\mathsf{H}}^{\pm}, u_{\mathsf{V}}^{\pm} \in \mathbb{R}^2$.

Then TV regularization is equivalent to minimizing

$$x^T A^T A x - 2x^T A^T m + \begin{bmatrix} 1 \\ 1 \end{bmatrix}^T (u_{\mathsf{H}}^{+} + u_{\mathsf{H}}^{-} + u_{\mathsf{V}}^{+} + u_{\mathsf{V}}^{-}),$$

over non-negative vectors $x \in \mathbb{R}^4$ (why? check!).

Reduction of TV regularization to the quadratic problem $\arg\min_{z\in\mathbb{R}^{12}_+}\left\{\frac{1}{2}z^TQz + c^Tz\right\}$ with $Ez = b$

So we aim to minimize $\frac{1}{2}z^TQz + c^Tz$ with

$$z = \begin{bmatrix} x \\ u_{\mathrm{H}}^{+} \\ u_{\mathrm{H}}^{-} \\ u_{\mathrm{V}}^{+} \\ u_{\mathrm{V}}^{-} \end{bmatrix} \in \mathbb{R}^{12}_{+},$$

$$Q = \begin{bmatrix} 2A^TA & 0 & \dots & 0 \\ 0 & 0 & \dots & 0 \\ \vdots & & \ddots & \vdots \\ 0 & \dots & & 0 \end{bmatrix}, \qquad c = \begin{bmatrix} -2A^Tm \\ 1 \\ \vdots \\ 1 \end{bmatrix}.$$

Explicit form of the equality constraint, slide 1/2

The equality constraint $Ez = b$ is needed for enforcing the identities $L_\mathsf{H} x - u_\mathsf{H}^+ + u_\mathsf{H}^- = 0$ and $L_\mathsf{V} x - u_\mathsf{V}^+ + u_\mathsf{V}^- = 0$.

Since

$$z = \begin{bmatrix} x \\ u_\mathsf{H}^+ \\ u_\mathsf{H}^- \\ u_\mathsf{V}^+ \\ u_\mathsf{V}^- \end{bmatrix} \in \mathbb{R}^{12},$$

we have

$$Ez = \begin{bmatrix} L_H & -I & I & 0 & 0 \\ L_V & 0 & 0 & -I & I \end{bmatrix} z = 0.$$

Explicit form of the equality constraint, slide 2/2

Finally we get

$$E = \left[\begin{array}{rr|rr|rr|rr|rr|rr} 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & -1 & 0 & -1 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 & 1 \end{array}\right]$$

and

$$b = \left[\begin{array}{c} 0 \\ 0 \\ 0 \\ 0 \end{array}\right].$$

Implementation in Matlab: preliminaries

```
% Construct the 2x2 pixel image target as a vertical vector
target = [2;2;6;7];

% Construct the measurement matrix
A = [1 0 1 0; 0 1 0 1; 1 1 0 0; 0 0 1 1];

% Compute an ideal X-ray measurement
m = A*target;

% Record the size of the unknown. Here M*M=n,
% since the unknown is a MxM pixel image.
n = 4; M = 2;
```

Regularization parameter

We are actually considering the TV regularization problem in a restricted form. In general it is advisable to solve

$$x_{\mathrm{TV}(\alpha)} = \arg\min_{x \in \mathbb{R}_+^{2\times 2}} \{\|Ax - m\|_2^2 + \alpha\|L_H x\|_1 + \alpha\|L_V x\|_1\},$$

where $\alpha > 0$ is a regularization parameter. However, for now we keep $\alpha = 1$ and write the following in Matlab:

```
% Regularization parameter
alpha = 1;
```

```
% Construct prior matrices
LH = [1 0 -1 0; 0 1 0 -1];
LV = [1 -1 0 0; 0 0 1 -1];

% Construct the quadratic optimization problem matrix
Q = zeros(n+4*M*(M-1));
Q(1:n,1:n) = 2*A.'*A;

% Construct the vector h of the linear term
c = alpha*ones(n+4*M*(M-1),1);
c(1:n) = -2*(A.')*m(:);
```

```
% Construct input arguments for quadprog.m
Z = zeros(M*(M-1));
Aeq = [[LH,-eye(M*(M-1)),eye(M*(M-1)),Z,Z];...
       [LV,Z,Z,-eye(M*(M-1)),eye(M*(M-1))]];
beq = zeros(2*M*(M-1),1);
lb = zeros(n+4*M*(M-1),1);
ub = Inf(5*n,1);
AA = -eye(n+4*M*(M-1));
AA(1:n,1:n) = zeros(n,n);
iniguess = zeros(n+4*M*(M-1),1);
b = [repmat(10,n,1);zeros(4*M*(M-1),1)];
QPopt = optimset('quadprog');
QPopt = optimset(QPopt,'Algorithm',...
        'interior-point-convex', 'Display','iter');
```

```matlab
% Compute reconstruction using quadprog
z = quadprog(Q,c,AA,b,Aeq,beq,lb,ub,iniguess,QPopt);

% Pick out the reconstructed image
recn = z(1:n);

% Show the reconstruction in image format
reshape(recn,M,M)
```

```
>> reshape(recn,M,M))

ans =

    2.2500 6.2500
    2.2500 6.2500
```

Total variation regularization

$2\frac{1}{4}$	$6\frac{1}{4}$
$2\frac{1}{4}$	$6\frac{1}{4}$

	data penalty	1
+	prior penalty	8
=	total penalty	€9

Outline

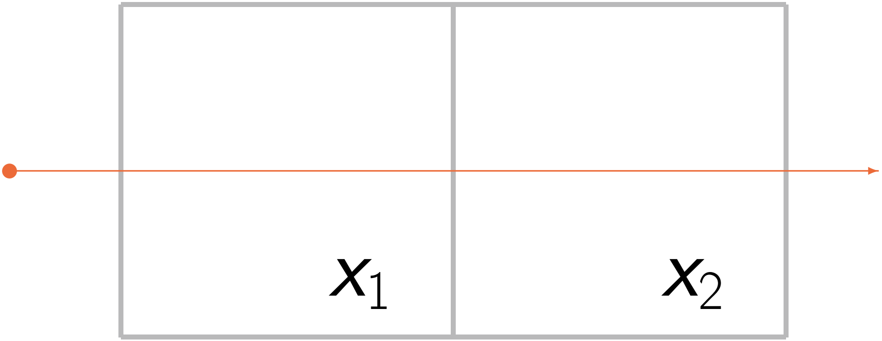
Third 2×2 exercise

Run the computation of the previous slide using the Matlab routine tomo2x2_TV_comp_quadprog.m in the Git repository https://github.com/ssiltane/BolognaWinterSchool2023
Note that you need the Optimization Toolbox.

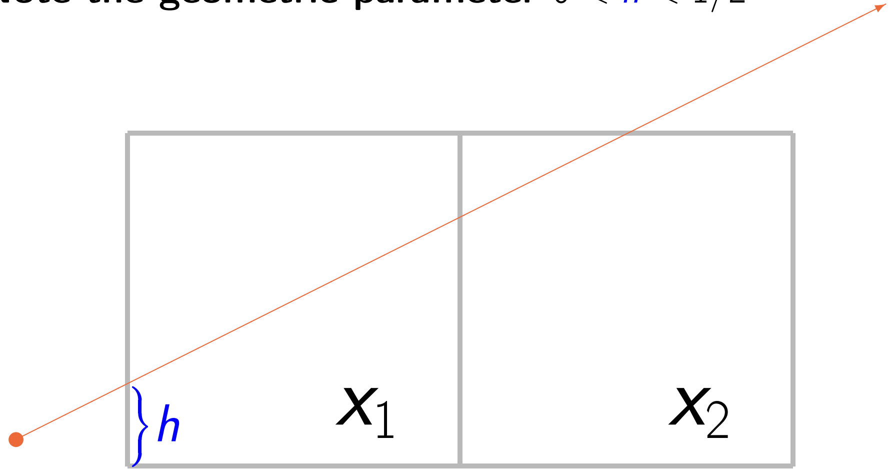
Then repeat the computation with several values of regularization parameter $\alpha > 0$. What choice of $\alpha > 0$ gives the smallest difference (measured in standard Euclidean norm of $\mathbb{R}^4$) between the true target and the regularized solution? Give the optimal α with the accuracy of two correct digits after the decimal point.

Outline

The first X-ray in our measurement travels horizontally



Second X-ray in the measurement has slope $1/2$. Note the geometric parameter $0 < h < 1/2$



First 1×2 exercise: construct measurement matrix

Assuming that the side length of pixel is one, write down the 2×2 matrix A_h modelling the measurement. (Some of the matrix elements may depend on $h > 0$.)

- Show that A_h is invertible for any $0 < h < 1/2$.
- What happens to $\det(A_h)$ when $h \to 0$? Why?
- We assume everywhere else that $0 < h < 1/2$. However, in this problem we step outside that assumption a bit. Is A_h invertible when $1/2 \leq h < 1$? How about the case $h \geq 1$?

Naive inversion

The measurement model is

$$A_h \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \end{bmatrix},$$

or $A_h x = m$ in short. Now assume that we have noisy data

$$\widetilde{m} = A_h x + \varepsilon.$$

Here $\varepsilon \in \mathbb{R}^2$ is a random noise vector. If A_h is invertible, we can attempt naive inversion $A_h^{-1}\widetilde{m}$. In the next exercise you will analyse this idea.

Second 1×2 exercise: ill-posedness of naive inversion

Naive reconstruction is an approximation of the unknown x, as we can see by this calculation:

$$A_h^{-1}\widetilde{m} = A_h^{-1}(A_h x + \varepsilon) = x + A_h^{-1}\varepsilon.$$

So we can bound the error by

$$\|A_h^{-1}\varepsilon\|_{\mathbb{R}^2} \leq \|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2}\|\varepsilon\|_{\mathbb{R}^2},$$

where $\|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2}$ is the operator norm of A_h.

- Compute the eigenvalues $\lambda_1(h) > 0$ and $\lambda_2(h) > 0$ of the matrix $A_h^T A_h$ numerically for a sequence of h values approaching zero. The numbers $s_j(h) = \sqrt{\lambda_j(h)}$ are called *singular values* of A_h. We order them so that $s_1 \geq s_2$.
- Now $\|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2} = 1/s_2(h)$. What happens to $\|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2}$ when $h \to 0$? What does that mean for the error bound?

Outline

First 2×2 exercise

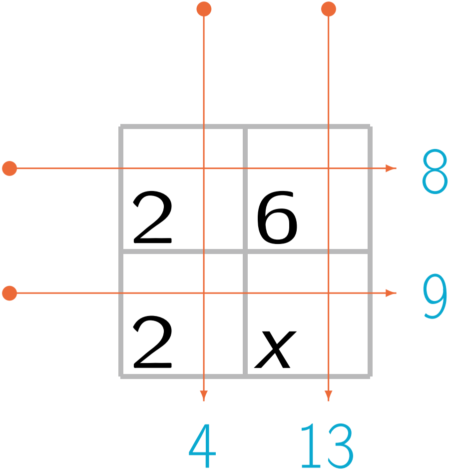
Determine the kernel of the measurement matrix

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix}.$$

How is the kernel related to "ghosts", or objects that are nontrivial but give zero measurement?

Second 2×2 exercise, slide 1/2

Assume that we know three pixel values and look for the fourth one, called x.



Second 2×2 exercise, slide 2/2

Take $\alpha = 1$. Write down the total variation penalty functional in the form

$$\widetilde{x} = \arg\min_{x \in \mathbb{R}} \{f(x)\}.$$

- Give the formula for f.
- Plot $f(x)$.
- At what points does f fail to be differentiable?
- Find the minimizing argument $\widetilde{x} \in \mathbb{R}$ approximately. You can either use brute-force forking or apply an optimization method.

Third 2×2 exercise

Run the computation of the previous slide using the Matlab routine tomo2x2_TV_comp_quadprog.m in the Git repository https://github.com/ssiltane/BolognaWinterSchool2023
Note that you need the Optimization Toolbox.

Then repeat the computation with several values of regularization parameter $\alpha > 0$. What choice of $\alpha > 0$ gives the smallest difference (measured in standard Euclidean norm of $\mathbb{R}^4$) between the true target and the regularized solution? Give the optimal α with the accuracy of two correct digits after the decimal point.

First 1×2 exercise: construct measurement matrix

Assuming that the side length of pixel is one, write down the 2×2 matrix A_h modelling the measurement. (Some of the matrix elements may depend on $h > 0$.)

- Show that A_h is invertible for any $0 < h < 1/2$.
- What happens to $\det(A_h)$ when $h \to 0$? Why?
- We assume everywhere else that $0 < h < 1/2$. However, in this problem we step outside that assumption a bit. Is A_h invertible when $1/2 \leq h < 1$? How about the case $h \geq 1$?

Second 1×2 exercise: ill-posedness of naive inversion

Naive reconstruction is an approximation of the unknown x, as we can see by this calculation:

$$A_h^{-1}\widetilde{m} = A_h^{-1}(A_h x + \varepsilon) = x + A_h^{-1}\varepsilon.$$

So we can bound the error by

$$\|A_h^{-1}\varepsilon\|_{\mathbb{R}^2} \leq \|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2}\|\varepsilon\|_{\mathbb{R}^2},$$

where $\|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2}$ is the operator norm of A_h.

- Compute the eigenvalues $\lambda_1(h) > 0$ and $\lambda_2(h) > 0$ of the matrix $A_h^T A_h$ numerically for a sequence of h values approaching zero. The numbers $s_j(h) = \sqrt{\lambda_j(h)}$ are called *singular values* of A_h. We order them so that $s_1 \geq s_2$.
- Now $\|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2} = 1/s_2(h)$. What happens to $\|A_h^{-1}\|_{\mathbb{R}^2\to\mathbb{R}^2}$ when $h \to 0$? What does that mean for the error bound?


SCIENCE
THE TRUE FANTASY